

**AFRL-VA-WP-TP-2006-316**

**SMOOTH FUNCTION MODELING FOR  
ON-LINE TRAJECTORY RESHAPING  
APPLICATION (PREPRINT)**

**Ajay Verma, Kaylan Vadakkevedu, Michael W. Oppenheimer, and  
David B. Doman**



**FEBRUARY 2006**

**Approved for public release; distribution is unlimited.**

**STINFO COPY**

**This work, resulting in whole or in part from Department of the Air Force contract FA8650-05-M-3505, has been submitted to AIAA for publication in the 2006 AIAA Guidance Navigation, and Control Conference Proceedings. If this work is published, AIAA may assert copyright. The United States has for itself and others acting on its behalf an unlimited, paid-up, nonexclusive, irrevocable worldwide license to use, modify, reproduce, release, perform, display, or disclose the work by or on behalf of the Government. All other rights are reserved by the copyright owner.**

**AIR VEHICLES DIRECTORATE  
AIR FORCE MATERIEL COMMAND  
AIR FORCE RESEARCH LABORATORY  
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542**

# NOTICE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the Air Force Research Laboratory Wright Site (AFRL/WS) Public Affairs Office (PAO) and is releasable to the National Technical Information Service (NTIS). It will be available to the general public, including foreign nationals.

PAO Case Number: AFRL/WS 06-0589, 28 Feb 2006.

THIS TECHNICAL REPORT IS APPROVED FOR PUBLICATION.

//Signature//

---

Michael W. Oppenheimer  
Electronics Engineer  
Control Design and Analysis Branch  
Air Force Research Laboratory  
Air Vehicles Directorate

//Signature//

---

Deborah S. Grismer  
Chief  
Control Design and Analysis Branch  
Air Force Research Laboratory  
Air Vehicles Directorate

//Signature//

---

Brian W. Van Vliet  
Chief  
Control Sciences Division  
Air Force Research Laboratory  
Air Vehicles Directorate

This report is published in the interest of scientific and technical information exchange and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

\*Disseminated copies will show “//Signature//” stamped or typed above the signature blocks

REPORT DOCUMENTATION PAGE					Form Approved OMB No. 0704-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. <b>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</b></p>						
1. REPORT DATE (DD-MM-YY) February 2006		2. REPORT TYPE Conference Paper Preprint		3. DATES COVERED (From - To) 04/01/2005– 02/15/2006		
4. TITLE AND SUBTITLE SMOOTH FUNCTION MODELING FOR ON-LINE TRAJECTORY RESHAPING APPLICATION (PREPRINT)				5a. CONTRACT NUMBER FA8650-05-M-3505		
				5b. GRANT NUMBER		
				5c. PROGRAM ELEMENT NUMBER 0605502		
6. AUTHOR(S) Ajay Verma and Kaylan Vadakkevedu (Knowledge Based Systems, Inc.) Michael W. Oppenheimer and David B. Doman (AFRL/VACA)				5d. PROJECT NUMBER A05G		
				5e. TASK NUMBER		
				5f. WORK UNIT NUMBER 0C		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Knowledge Based Systems, Inc. 1408 University Drive College Station, TX 77840 Control Design and Analysis Branch (AFRL/VACA) Control Sciences Division Air Vehicles Directorate Air Force Materiel Command, Air Force Research Laboratory Wright-Patterson Air Force Base, OH 45433-7542				8. PERFORMING ORGANIZATION REPORT NUMBER AFRL-VA-WP-TP-2006-316		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Vehicles Directorate Air Force Research Laboratory Air Force Materiel Command Wright-Patterson Air Force Base, OH 45433-7542				10. SPONSORING/MONITORING AGENCY ACRONYM(S) AFRL-VA-WP		
				11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S) AFRL-VA-WP-TP-2006-316		
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.						
13. SUPPLEMENTARY NOTES <p>This work, resulting in whole or in part from Department of the Air Force contract FA8650-05-M-3505, has been submitted to AIAA for publication in the 2006 AIAA Guidance Navigation, and Control Conference Proceedings. If this work is published, AIAA may assert copyright. The United States has for itself and others acting on its behalf an unlimited, paid-up, nonexclusive, irrevocable worldwide license to use, modify, reproduce, release, perform, display, or disclose the work by or on behalf of the Government. All other rights are reserved by the copyright owner.</p> <p>PAO Case Number: AFRL/WS 06-0589 (cleared February 28, 2006).</p> <p>This is a Small Business Innovation Research (SBIR) Phase II report. Paper contains color.</p>						
14. ABSTRACT <p>Online vehicle trajectory reshaping is desired for a class of autonomous air vehicles such as RLVs in order to avoid catastrophic failure when subjected to performance restricting damages and failures. An <i>Adaptive Trajectory Reshaping and Control</i> (ATRC) system is envisioned that responds to altered vehicle conditions by continuously retargeting and reshaping the reference RLV trajectory satisfying the feasibility constraints. On-line trajectory reshaping to determine a feasible reference trajectory is computationally a difficult problem for real time applications. ATRC is exploring the principles of vehicle dynamics inversion for on-line generation of feasible reference trajectory.</p>						
15. SUBJECT TERMS SBIR Report, trajectory reshaping, inverse dynamics, aerodynamic model						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT: SAR	18. NUMBER OF PAGES 22	19a. NAME OF RESPONSIBLE PERSON (Monitor) Michael W. Oppenheimer 19b. TELEPHONE NUMBER (Include Area Code) N/A	
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified				

# Smooth Function Modeling for On-Line Trajectory Reshaping Application

Ajay Verma<sup>\*</sup>, Kalyan Vadakkeveedu<sup>†</sup>  
*Knowledge Based Systems, Inc.*

Michael W. Oppenheimer<sup>‡</sup> and David B. Doman<sup>§</sup>  
*AFRL/VACA*

Online vehicle trajectory reshaping is desired for a class of autonomous air vehicles such as RLVs in order to avoid catastrophic failure when subjected to performance restricting damages and failures. An *Adaptive Trajectory Reshaping and Control*<sup>1</sup> (ATRC) system is envisioned that responds to altered vehicle conditions by continuously retargeting and reshaping the reference RLV trajectory satisfying the feasibility constraints. On-line trajectory reshaping to determine a feasible reference trajectory is computationally a difficult problem for real time applications. ATRC is exploring the principles of vehicle dynamics inversion for on-line generation of feasible reference trajectory. Two essential components for generating reference trajectory for air-vehicles using “inverse dynamics” methodology are aerodynamic model of the vehicle that is representative of the current state of the vehicle, and a framework for modeling the vehicle trajectory. Physics based modeling such as DATCOM allows fast computation of aerodynamic coefficients for given flight points and the results can be stored in a tabular form. However, for efficient real-time trajectory reshaping application, it is desired to represent aerodynamic coefficients in smooth functional form that are governed by few parameters. Similarly, trajectories must also be represented by smooth functions. In this paper we present modeling of smooth functions using a set of basis functions that are suitable for trajectory reshaping of the air vehicles. A desirable feature for function modeling is the easy imposition of boundary as well as mid point constraints in the function using small number of parameters without limiting the scope of the function. In this paper we present a design of polynomial based set of constrained orthonormal basis functions in one and two dimension.

## I. Introduction

The large potential for space utilization is not being exploited as it is currently inhibited by the huge cost of launching operations. The benefit of advance space utilization can be greatly increased by making space utilization more affordable. The Reusable Launch Vehicle (RLV) programs are targeted towards affordable space utilization. However, to maintain the economical viability of RLVs, it is important to enhance operations safety and reliability by providing the RLV the capability to respond to various uncertainties and emerging emergency situations. Responding to an uncertain environment after a damage/failure presents many tough technical problems for this class of vehicle. These problem manifests in the following challenges that must be overcome: First, to adequately determine and model the dynamic

---

<sup>\*</sup> Senior Researcher, 1408 University Dr, College Station, TX-77840, Senior Member AIAA, [averma@KBSI.com](mailto:averma@KBSI.com).

<sup>†</sup> Researcher, 1408 University Dr, College Station, TX-77840, Member AIAA, [kvadakkevedu@KBSI.com](mailto:kvadakkevedu@KBSI.com).

<sup>‡</sup> Electronics Engineer, 2210 Eighth Street, Bldg. 146, Rm. 305, WPAFB, OH 45433-7531, Member AIAA, Member AIAA, [Michael.Oppenheimer@wpafb.af.mil](mailto:Michael.Oppenheimer@wpafb.af.mil)

<sup>§</sup> Senior Aerospace Engineer, 2210 Eighth Street, Bldg. 146, Rm. 305, WPAFB, OH 45433-7531, Associate Fellow AIAA, [David.Doman@wpafb.af.mil](mailto:David.Doman@wpafb.af.mil)

characteristics of the vehicle in the altered state after a damage/failure; second, to estimate the new constraints and limitation(s) of the vehicle; third, to adopt and reconfigure the command, control and guidance of the vehicle to the modified system dynamics; and fourth, to design and plan a new feasible path with respect to the end goal maximization. A high percentage of such damage/failure cases leave the vehicle in an uncontrolled and uncertain environment with a highly likely probability of ultimately entering a state for catastrophic failure. The high cost of loss resulting from catastrophic failures, has prompted researchers in the direction of developing technologies to assist in minimizing such failures.

Damage to a vehicle or a sub-system failure may result in modification of the applicable trajectory constraints and/or the dynamical behavior of the system, consequently making the previously designed reference trajectory infeasible. An acceptable trajectory for a dynamical system is a solution of a two-point boundary value problem for a set of governing differential equation of motion. The real world systems such as air vehicles are highly non-linear systems and impose a set of constraints on the trajectory variables as well as control variables. In inverse dynamics approach, a trajectory is specified first, which results in solving a set of algebraic equations, yet strictly satisfying the non-linear differential equations of a non-flat system.

In this paper first we introduce the architecture of an Adaptive Trajectory Reshaping and Control (ATRC) system for general class of RLV system, which is based on the principles as described in Ref. 1. Next we discuss the general inverse dynamics approach for trajectory reshaping of RLVs and the motivation for functional modeling. For real time trajectory determination, there are two types of functions that must be modeled. One set of function is needed to describe the vehicle trajectory. The functions for the spatial coordinates of the vehicle constitute a vehicle trajectory. These trajectory functions are normally independent in one variable. The second set of functions that must be modeled in real time is to approximate the modified aerodynamic constraints. The aerodynamic constraints are modeled by defining aerodynamic coefficients for the flight envelope of the vehicle. Mostly, these functions are two dimensional with *Mach* and *angle of attack* being the independent parameters. A general approach to model a function is to parameterize the function and then determine the parameters that satisfy any governing constraints to a satisfactory level. In section IV, we describe a novel approach to design a set of basis functions with a class of constraints built in, that reduces the complexity of the solution and results in better approximations.

## **II. Adaptive Trajectory Reshaping and Control (ATRC) System**

The ATRC system enhances RLV capability to avoid catastrophic failure when subjected to performance restricting damages and failures. The overall goal of ATRC translates into specific requirements for design and development of functionalities related to adaptable and reconfigurable command, control, and guidance system for the RLVs. and real time solution techniques for an.

Figure 1 shows the general architecture of the envisioned ATRC system for RLVs. Note that the above structure is specific to longitudinal motion of the vehicle, however, it can be easily extended to six degree of freedom. The main components of the envisioned ATRC system requires:

1. On-line system identification that includes parameter estimation and parameter projection for constraint boundary determination. The constraint boundaries influence the trajectory reshaping of the vehicle.
2. Real time trajectory determination for reshaping reference trajectory under feasibility constraints.
3. Adaptive, closed loop control and guidance system for reference trajectory tracking.

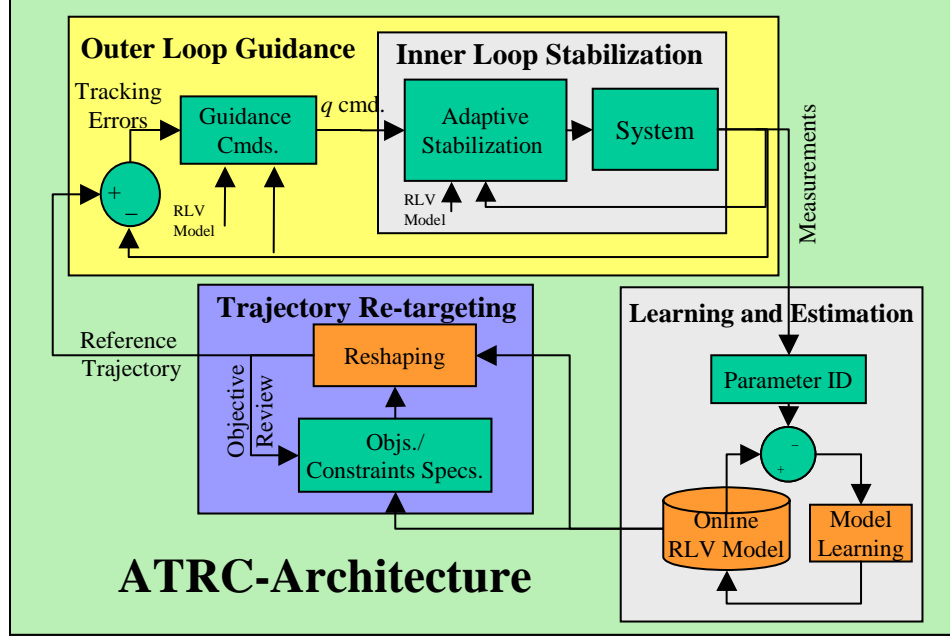


Figure 1. Architecture of ATRC

### III. Inverse Dynamics Approach for Trajectory Generation

The advantage of inverse dynamics approach is that one can avoid integration of differential equations altogether. The main difference is that we parameterize the trajectory itself and then uses numerical techniques for solving these parameters that minimizes the objective function and other inequality constraints. Once a smooth trajectory is specified, the time derivatives are also fixed and hence the differential equations become simple algebraic equations.

#### A. Reference Trajectory Design

To determine a trajectory for a non-linear dynamic system, a solution must be found that satisfies the set of differential equations governing the dynamics of the system. Further, the trajectory solution should not violate some non-linear constraints, which limits the operation capability of the system. For an aircraft, the constraints arise due to several factors such as the limitations on *angle of attack*, *load factor*, *aerodynamic heating*, and *actuator saturation*.

There are two primary approaches for trajectory generation and these have been classified in the literature [8] as the “integral approach” and the “differential approach.” In any approach, where generation of a trajectory involves the integration of the equations of motion, this approach is classified as the “integral approach.” In a differential approach, an assumed functional form for trajectory is differentiated to obtain algebraic functions for the higher derivatives, which are required to impose constraints on the control inputs for the “inverse dynamics” solution. There are various applications where inverse dynamics have been used, such as spacecraft trajectories and path planning in robotics [9] and overhead cranes [10]. Historically, the inverse dynamics approach has been used for “differentially flat” systems. A system is “differentially flat” [see 11, 12] if there exists a set of outputs, known as “flat outputs,” such that there is a one-to-one correspondence between the trajectories of flat outputs and the full state and control inputs of the system. In our approach, we use an inverse dynamics approach for determining trajectory as this allows us to solve algebraic equations instead of integrating ODEs. With the inverse dynamics approach for aircraft trajectories, a problem arises due to inherent under-actuation in most of the aircrafts. For a six-degree of freedom aircraft, there are normally four controls: thrust, elevator, aileron, and rudder. Ref. [7] defines a novel trajectory generation scheme, which uses pseudo forces for inverse dynamic computation. In this paper we concentrate on function modeling approach that to make the inverse dynamic solution approach more efficient.

## B. Inverse Dynamics Approach

Assume  $x$  be the state vector for a vehicle and  $x = x(\tau)$  represent the trajectory of the vehicle. The governing equations of motion for an air vehicle can be represented in the following form

$$\dot{x} = \begin{bmatrix} C(v(x)) \\ x \\ u \end{bmatrix} = C_1(v(x)) + C_2(v(x))x + C_3(v(x))u, \quad (1)$$

where  $v(x)$  are the independent parameters defined in terms of the subset of vehicle states, and  $C_i(v(x))$  are the non-linear coefficient functions in independent parameters.

For example, the pitch dynamics of a vehicle for longitudinal dynamics can be written as

$$I\dot{q} = C_{m0}(M, \alpha) + C_{m_q}(M, \alpha)q + C_{m_{\dot{q}}}(M, \alpha)\delta e. \quad (2)$$

The inverse dynamic solution for Eq. (1) can be written as

$$u = C_3^{-1}(v(x))(\dot{x} - C_1(v(x)) - C_2(v(x))x) \quad (3)$$

For a real time solution, in presence of uncertainty due to damage to the vehicle requires modeling of the trajectory  $x = x(\tau)$  and estimated models of the coefficient functions  $C_i(v(x))$ . In the following section we will address one dimensional trajectory modeling and two-dimensional coefficient functional representation in parametric form.

## IV. One-Dimensional Function Modeling with Constraints using Basis Functions

Often we need to estimate or model non-linear functions that must satisfy some constraints. For example, the inverse dynamic solution approach requires parameterization of multiple system outputs variables or the trajectory variables with some constraints imposed on it. Note that the feasible trajectory must also satisfy the governing differential equations. The trajectory parameters are determined using an optimization process that minimizes the error between trajectory variables and the solution of the governing equations. Note that if some of the trajectory constraints, such as the boundary constraints are forced in the trajectory parameterization itself, it helps in reducing the complexity of the optimization problem. Verma et al [4,7] presented the following approach to represent trajectory that is useful to impose boundary and inpoint constraints. To prescribe a smooth trajectory functional for a specific trajectory coordinate, the position coordinate is represented by a twice differentiable, smooth function so that velocity and acceleration can be uniquely defined. The path for an individual position coordinate is defined as a function of normalized time  $\tau = t/T_f$ , where  $T_f$  is the total time for the maneuver. An individual coordinate trajectory is structured to have two parts, a *base trajectory* and a *perturbation* of the base trajectory. If  $P(\tau)$  is the  $i^{th}$  coordinate, it is chosen to be of the form

$$P(\tau) = P_0(\tau) + P_1(\tau) \sum_{j=1}^l \alpha_j \varphi_j(\tau) \quad (4)$$

Here  $P_0(\tau)$  is the base trajectory function, which is chosen such that it satisfies any boundary or mid point constraints for the the trajectory. One example approach  $P_0(\tau)$  could be chosen as a minimum degree polynomial spline that satisfies the desired constraint conditions. The base trajectory by itself may not be a true solution of the governing equations and hence an infeasible trajectory. To make the trajectory feasible, we add a second term on the right hand side of the trajectory equation. The perturbation term must be designed to ensure that the overall trajectory function  $P(\tau)$  is the solution of the governing equations. Perturbation term is defined using a set of ortho-normal basis functions  $\varphi_j(\tau)$  that are modified by the term  $P_1(\tau)$ . The term  $P_1(\tau)$  is a weight polynomial constraining the perturbation term to contribute zero value for the already satisfied desired constraints by base trajectory  $P_0(\tau)$ . For example  $P_1(\tau)$  can be chosen as

$$P_1(\tau) = (\tau - a)^p (\tau - b)^q, \quad (5)$$

where  $p$  and  $q$  are integers, usually less or equal to 3. The integers  $p$  and  $q$  depend upon the highest order condition match required at times  $\tau = a$  and  $\tau = b$  respectively. For example, when boundary

conditions at  $\tau=0$  and  $\tau=1$  up to acceleration level must be imposed on the trajectory function,  $P_1(\tau)$  can be chosen as  $\tau^3(\tau-1)^3$ . The drawback of this approach is that the after basis functions  $\varphi_j$  are modified by  $P_1(\tau)$ , it loses the orthogonality, which makes it computationally harder to determine the coefficients for the trajectory solution. In the next section we present an approach to design orthogonal polynomial basis functions with inherent desired constraints.

## V. Constraint Orthogonal Polynomial Basis Functions

The choice of basis functions ultimately influences the complexity of the modeling of a desired function or behavior. It is well known that a sub set of a complete set of basis functions can approximate any given function to a desired accuracy by choosing a sufficient number of basis functions elements in the sub set. However, if a function to be approximated must meet certain constraints, it presents various problems. First, a finite set of basis function may not ensure the constraints satisfaction on the function to be approximated. Second, a large number of basis functions may be required for the satisfactory constraint approximation. Now, if we can create a set of basis functions, where all the basis functions satisfy the given constraints, any linear combination of those functions will also satisfy the given constraints. With this motivation we developed an approach to design a constraint orthogonal polynomial basis functions. To explain our approach, we first present a way for generating one-dimensional unconstraint orthogonal polynomial based basis functions.

### A. One Dimensional Basis Functions

Let  $\Phi_i(x)$  represent the 1-D  $i^{\text{th}}$  function of the orthonormal polynomial basis functions given as:

$$\Phi_i(x) = \sum_{k=0,i} a_k x^k, \quad i = 0, 1, \dots \quad (6)$$

Notice that the  $i^{\text{th}}$  basis function has  $i+1$  polynomial coefficients to be determined. We impose the constraints on the functions to determine the coefficients to ensure orthogonality and normalization. Defining the inner product of two functional as

$$\langle \Phi_i(x), \Phi_k(x) \rangle = \int_0^1 \Phi_i(x) \Phi_k(x) dx. \quad (7)$$

The norm and the orthogonality conditions can be obtained as:

$$\int_0^1 (\Phi_i(x))^2 dx = 1, \quad \text{Normalizing condition.} \quad (8)$$

$$\int_0^1 \Phi_i(x) \Phi_k(x) dx = 0, \quad k = 0, 1, \dots, i-1, \quad \text{Orthogonality condition.} \quad (9)$$

The orthogonality condition gives  $i$  constraints while an additional constraint from normalization ensures that all the coefficients can be obtained.

For automatically generating the basis set we used the following recursive approach. Assuming that we already know  $\Phi_k, k=0, i-1$ . Define an arbitrary polynomial  $f_i(x)$  of degree  $i$ . For  $k=0$  to  $i-1$  we modify  $f_i(x)$  as

$$f_i^{k+1}(x) = f_i^k(x) - \langle \Phi_k(x), f_i^k(x) \rangle \Phi_k(x) \quad (10)$$

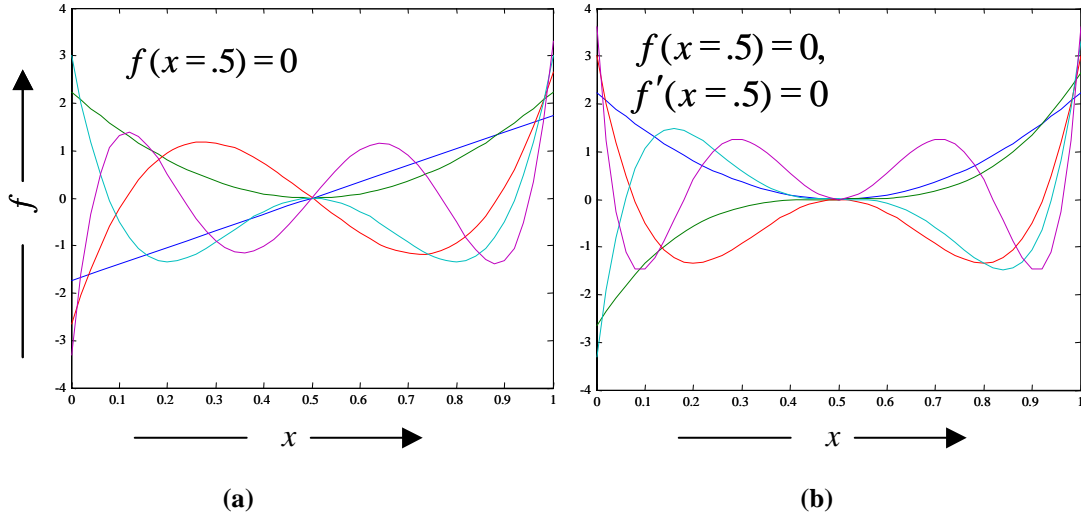
The  $i^{\text{th}}$  basis function  $\Phi_i(x)$  is now given as

$$\Phi_i(x) = f_i^i(x) \quad (11)$$

For constrained orthogonal polynomial function, Eq. (6) is modified as follows.

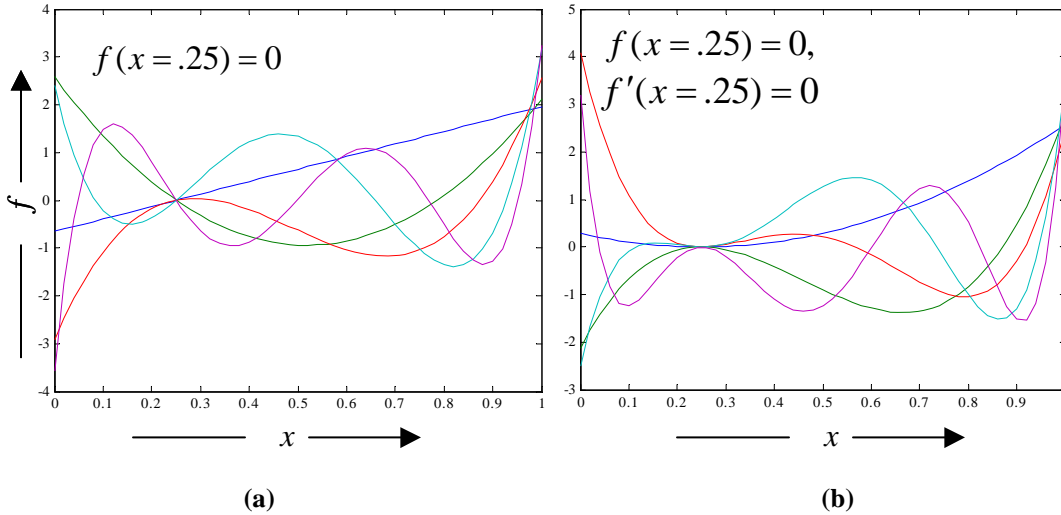
$$\Phi_i(x) = \psi(x) \sum_{k=0,i} a_k x^k, \quad i = 0, 1, \dots, \quad (12)$$

where,  $\psi(x)$  imposes a desired constraint on the basis function. Figure 2 and Figure 3 show some examples of a class of orthogonal basis functions that incorporate the given constraints. The example presents normalized orthogonal basis functions in the range 0 to 1. In



**Figure 2. Constraint Polynomial Basis Functions. (a) Basis Function are Constraint to be Zero at  $x = 0.5$ . (b) Both, Function and its Derivative are Constrained to be Zeros at  $x = 0.5$**

Figure 2(a), the value of the function is constraint to be zero mid way, i.e.  $f(x)|_{x=0.5} = 0$ . Figure 2(b) adds an additional constraint on the slope of the curve or the first derivative of the function given as  $df(x)/dx|_{x=0.5} = 0$ . Note that the first set of constraint results in anti-symmetric orthogonal basis functions as can be seen in Figure 2(a), while the second set of constraint results in symmetric orthogonal basis functions as shown in Figure 2(b). In Figure 3, the application of the constraint has been shifted to the quarter point, i.e.  $x = 0.25$ . In this case, the basic nature of orthogonal basis functions remains the same, albeit with a loss of symmetry. Figure 2 and Figure 3 demonstrate the ability to generate a range of readily available sets of *constraint orthogonal polynomial basis functions*.



**Figure 3. Constraint Polynomial Basis Functions. (a) Basis Function are Constraint to be Zero at  $x = 0.25$ . (b) Both, Function and its Derivative are Constrained to be Zeros at  $x = 0.25$**

## B. Ortho-Normal Polynomials Basis Functions in Two-Dimensions

In this section we demonstrate an approach to produce a set of polynomial orthonormal basis functions in two dimensions, represented by  $x$  and  $y$  with maximum degree of  $N, M$  respectively. First we introduce the vector space of two-dimensional polynomials and then the algorithm to generate orthonormal basis vectors.

### 1. Matrix Representation of 2-D Polynomials

Let  $\mathbf{P}_{N,M}$  be a set of polynomials where  $P(x, y) \in \mathbf{P}_{N,M}$  is a polynomial in  $x$  and  $y$  of degree  $N$  and  $M$  respectively. Let the coefficients be  $a_{ij}$ .

$$P(x, y) = a_{NM}x^N y^M + a_{N-1M}x^{N-1}y^M + \cdots + a_{0M}x^0 y^M + a_{N-1M}x^{N-1}y^M + \cdots + a_{00}x^0 y^0,$$

or

$$P(x, y) = \sum_{i=1}^N \sum_{j=1}^M a_{i,j} \cdot x^{N-i} \cdot y^{M-j}. \quad (13)$$

Let us represent the polynomial coefficients in matrix form as,

$$A = \begin{bmatrix} a_{NM} & a_{N-1M} & a_{N-2M} & \cdots & a_{0M} \\ a_{NM-1} & a_{N-1M-1} & a_{N-2M-1} & \cdots & a_{0M-1} \\ \vdots & & \ddots & & \vdots \\ a_{N0} & a_{N-10} & a_{N-20} & \cdots & a_{00} \end{bmatrix}_{M+1, N+1}. \quad (14)$$

Then the polynomial  $P(x, y)$  can be represented in matrix form as,

$$P(x, y) = X^T A Y, \quad (15)$$

where,

$$X = \begin{bmatrix} x^N \\ x^{N-1} \\ \vdots \\ 1 \end{bmatrix}_{N+1}, \text{ and } Y = \begin{bmatrix} y^M \\ y^{M-1} \\ \vdots \\ 1 \end{bmatrix}_{M+1}. \quad (16)$$

### 2. Representation of a 2-D Polynomial Inner Product Vector Space

Let  $\mathbf{P}_{N,M}$  be a set of polynomials in  $x$  and  $y$  of degrees up to  $N$  and  $M$  respectively. The polynomial set  $\mathbf{P}_{N,M}$  defines a polynomial vector space over  $\mathbf{R}$  (set of real numbers) with

$$\Theta = \{\theta_{i,j} \mid \theta_{i,j} = x^i y^j\}_{0 \leq i \leq N, 0 \leq j \leq M} \quad (17)$$

as the basis vectors that satisfy the basic properties of associativity, commutativity, existence of identity and inverse, and properties of scalar multiplication. In order to define orthogonality, we must define an *inner product space*. An *inner product space* is a *vector space* with an additional structure of *inner product*. We define the *inner product* on the 2-D polynomial vector space as,

$$\langle P(x, y), Q(x, y) \rangle = \int_0^1 \int_0^1 P(x, y) \otimes Q(x, y) dx dy, \quad P(x, y), Q(x, y) \in \mathbf{P}_{N,M}. \quad (18)$$

The polynomial function multiplication  $\otimes$ , inside the double integrals can be implemented in MATLAB using 2D convolution of the corresponding coefficient matrices (see function conv2).

Our goal is to construct an orthonormal basis for the polynomial vector space  $\mathbf{P}_{N,M}$ . As a first step we compute an orthonormal basis set of vectors using the Gram-Schmidt procedure. Then we compute the coefficients of a polynomial function  $P(x, y)$  in the new transformed space with orthonormal basis.

### 3. The Gram-Schmidt Procedure

If  $\mathbf{V}$  is a set of  $L$  linearly independent vectors, then the Gram Schmidt procedure can be used to find set  $\mathbf{W}$  of  $L$  orthonormal vectors that span the same vector space as  $\mathbf{V}$ . In the case of the 2D polynomial vector space, we have a set of  $(N+1) \times (M+1)$  linearly independent vectors  $\Theta = \{\theta_{i,j}\}$  that span the space of  $N \times M$  degree polynomials. We can apply Gram-Schmidt procedure to this vector space to compute the set of  $(N+1) \times (M+1)$  orthonormal vectors  $\Phi = \{\phi_{ij}\}$ .

$$\phi_{ij} = \frac{\varphi_{ij}}{\|\varphi_{ij}\|},$$

where,

$$\begin{aligned} \varphi_{00} &= 1, \\ \varphi_{01} &= x^1 y^0 - \langle x^1 y^0, \phi_{00} \rangle \phi_{00}, \\ \varphi_{02} &= x^2 y^0 - \langle x^2 y^0, \phi_{00} \rangle \phi_{00} - \langle x^2 y^0, \phi_{01} \rangle \phi_{01}, \\ &\vdots \\ \varphi_{ij} &= x^j y^i - \langle x^j y^i, \phi_{j-1,i} \rangle \phi_{j-1,i} - \langle x^j y^i, \phi_{j-2,i} \rangle \phi_{j-2,i} - \dots - \langle x^j y^i, \phi_{0,0} \rangle \phi_{0,0} \\ \text{and, } \|\varphi_{ij}\| &= \sqrt{\langle \varphi_{ij}, \varphi_{ij} \rangle}. \end{aligned}$$

Once we have the transformation from  $\Theta$  to  $\Phi$ , we can transform any given vector (polynomial in this case) to the new orthonormal basis and compute the new set of coefficients, by taking the projection of the given vector (polynomial) onto each of the new basis vectors.

The polynomial vector  $P(x, y)$  can be represented in a new basis with coefficients  $\{\alpha_{i,j}\}$ , where

$$\alpha_{ij} = \text{projection}(P(x, y), \phi_{ij}) = \langle P(x, y), \phi_{ij} \rangle.$$

### C. Constrained Orthonormal Basis Set for 2-D Functions

Many times we would like to represent a class of functions in terms of basis functions that must satisfy certain constraints. For an efficient representation, we would like to construct a set of basis functions satisfying the given constraints in addition to the orthonormality condition. In this report we restrict our discussion to three types of constraints; constraints on the function *value*, its *first derivative* or *both*. In the following we describe the generation of 2-D constrained basis functions.

- (i) Constraint only on the value of the function at a given point. I.e.,  $C(x, y) = 0$ . To introduce the constraints, first we modify the polynomial basis set  $\Theta$  by multiplying each basis function  $\theta_{i,j}(x, y)$  by  $C(x, y)$  to form a new constrained basis set  $\Theta_C$  given as

$$\Theta_C = \{\theta_{i,j}(x, y) \otimes C(x, y)\}.$$

Note that the basis set  $\Theta_C$  is not yet orthonormal. The constrained orthonormal basis set  $\Phi_C$  is constructed using the Gram-Schmidt procedure as discussed in the previous section. See Figure 4 for an example of the orthonormal basis functions generated.

- (ii) Constraint on the value and derivative along a curve  $C(x_0, y_0) = 0$ . I.e.  $C(x_0, y_0) = 0$  and  $C'(x_0, y_0) = 0$ .  $C'$  is the derivative in the direction orthogonal to the curve  $C(x_0, y_0) = 0$  in the  $x$ - $y$  plane. To introduce the constraints, we first define  $C(x, y)^2 = C(x, y) \otimes C(x, y)$ . Next we

modify the polynomial basis set  $\Theta$  by multiplying each basis function  $\theta_{i,j}(x,y)$  by  $C(x,y)^2$  to form a new constrained basis set  $\Theta_C$  given as

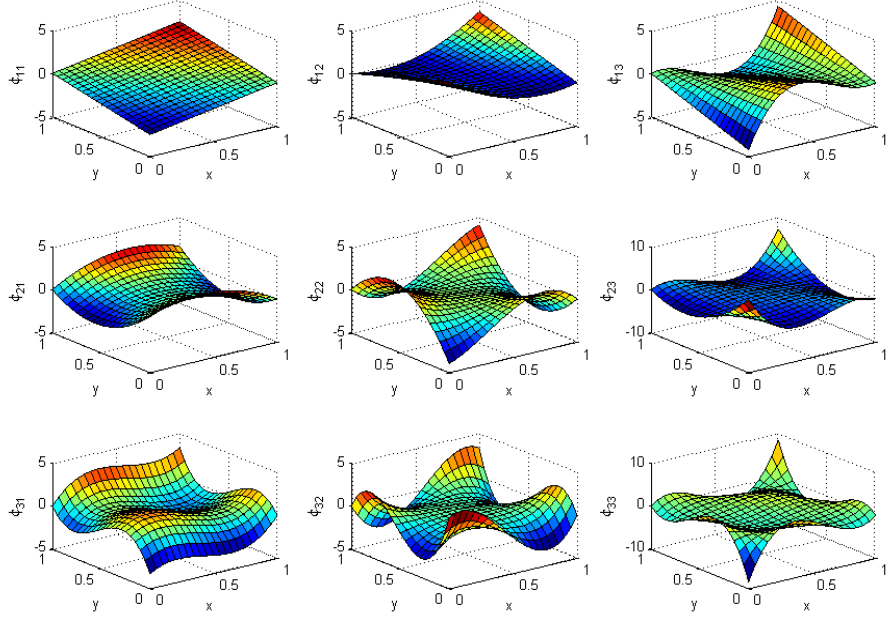
$$\Theta_C = \{\theta_{i,j}(x,y) \otimes C(x,y)^2\}.$$

The constrained orthonormal basis set  $\Phi_C$  is constructed using the Gram-Schmidt procedure as discussed in the previous section. See Figure 5 for an example of the orthonormal functions generated.

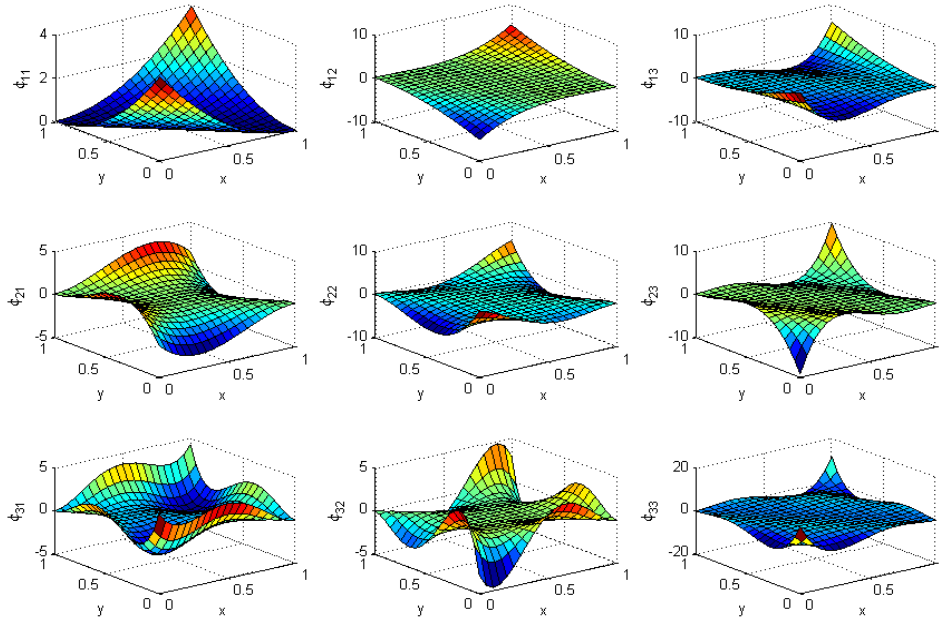
- (iii) Constraint only on the derivative of the function. We then modify the given polynomial  $\Theta$  as,

$$\Theta_C = \{\theta_{0,0}(x,y)\} \cup \{\Theta - \{\theta_{0,0}(x,y)\}\} \otimes C(x,y)^2.$$

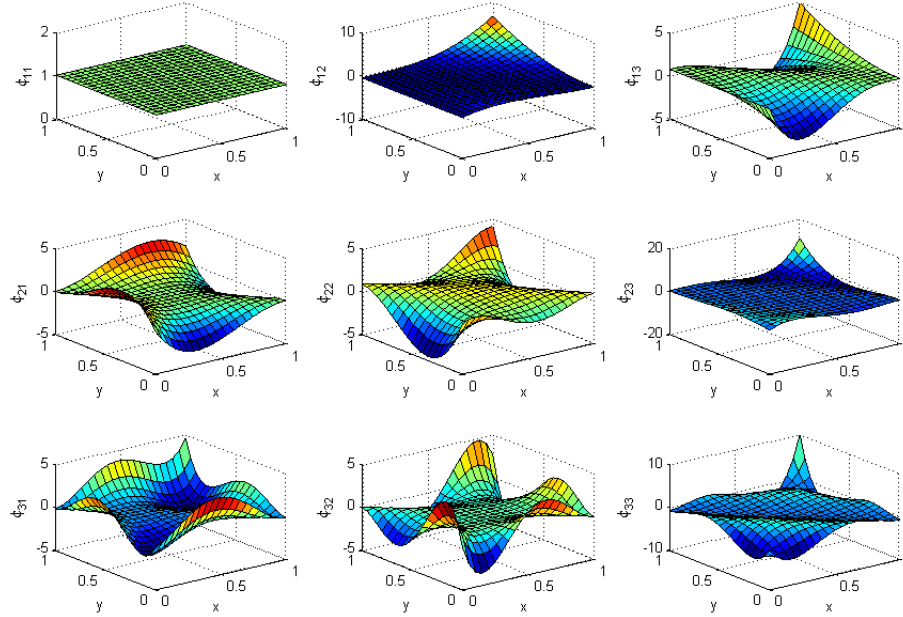
Figure 6 gives an example of orthonormal functions generated with constraints only on the derivatives.



**Figure 4. Orthonormal basis function set of polynomials in  $x$  and  $y$  of degrees 2 and 2, with the constraint that at  $x+y = 1$ , the value of the function is zero.**



**Figure 5. Orthonormal basis function set of polynomials in  $x$  and  $y$  of degrees 2 and 2, with the constraint that at  $x+y = 1$ , the value and derivatives of the function are zero.**



**Figure 6. Orthonormal basis function set of polynomials in  $x$  and  $y$  of degrees 2 and 2, with the constraint that at  $x+y = 1$ , the derivative of the function is zero.**

## VI. Aerodynamic Coefficient Function Estimation

An important goal of the ATRC system is the adaptive reshaping of the RLV trajectory in the presence of altered dynamic characteristics of the vehicle when unexpected damage occurs in the various operating scenarios. Any damage to a vehicle that has an impact on the external shape of the vehicle, or that creates an impediment in normal functioning of the control surfaces, results in alteration of the vehicle's aerodynamic characteristics. Figure 9 and Figure 10 show few examples of the Pitch moment coefficient variation in the presence of various damage scenarios. Since the aerodynamic behavior of the vehicle is captured in aerodynamic coefficients that are used for the design of vehicle control and trajectory planning, it becomes mission critical to adapt reference trajectory for the altered vehicle dynamics. Hence we need an approach to build a smooth on-line aerodynamic model. Physics based modeling such as DATCOM allows fast computation of aerodynamic coefficients for given flight points and the results can be stored in a tabular form. However, for efficient real-time trajectory reshaping application, it is desired to represent aerodynamic coefficients in smooth functional form that are governed by few parameters. In this section, we present a piecewise continuous and smooth function model in two dimensions.

### A. Physic Based Modeling for Aerodynamic Modeling

Given the geometry of the vehicle, very good estimations of the aerodynamic coefficients can be generated based on Physics based modeling in a very small time. For example, if damage on the vehicle is characterized adequately, the DATCOM technology allows specification of the altered geometry of the vehicle to compute the corresponding aerodynamic coefficients at desired points of the flight envelope. In our approach, we generate a large number of data points for the aerodynamic coefficients and then fit a piecewise continuous and smooth function to create a functional model for vehicle aerodynamics. The functional form is later used for trajectory reshaping.

### B. Finite Element Function Approximation

We formulate the aerodynamic coefficient function with a set of parameters. First we show our approach for piecewise continuous and smooth one-dimensional function, which is later extended to two-dimensions. We used a finite element modeling approach so that the approximation function would capture local variations in an efficient manner. Ref. 20,21 demonstrates the use of finite element piecewise approximation for mapping geopotential. Ref. 4,7 applied the technique for aerodynamic coefficients representation. First, the argument space of the function is divided to form a grid with one control point at every grid point. At each control point we use a local polynomial function that is determined using a weighted, least square method from a given set of nominal data generated from DATCOM.

#### 4. One-Dimensional Finite Element Approximation

The scope of each local polynomial centered at the control point lies in between the adjacent grid points (see Figure 7). Notice the overlapping of local polynomials, which helps in obtaining a smooth function over the entire range. Once local approximations are determined, a smooth global approximation is obtained as a weighted combination of the local approximations. The smoothness of the approximation function implies that the function, as well as its first derivative, is continuous. Notice that grids are not restricted to be equidistant. To capture nonlinearity effectively, more control points should be placed near high non-linearity. Figure 7 uses a second order polynomial function for local approximations.

Once local approximations are attained, a smooth global approximation is obtained as a weighted combination of the local approximations. A highlight of this Finite Element approximation is that it preserves the local function value and its first derivative at its control point. This is achieved by a smooth weighting function that gracefully goes from unity to zero, from one control point to another, without contributing to a first derivative at both control points. The weighting function for the two overlapping local approximation curves is given as

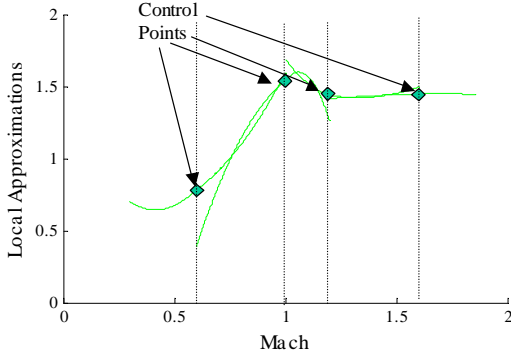
$$W_1(\bar{q}) = (1 - \bar{q})^2(1 + 2\bar{q}), \quad W_2(\bar{q}) = 1 - W_1(\bar{q}),$$

where  $\bar{q}$  is the normalized coordinate with the origin at the *first* control point. Note that the weighting function ensures that  $W_1(0)=1$ ,  $W_2(0)=0$  at the first control point and similarly,  $W_1(1)=0$ ,  $W_2(1)=1$

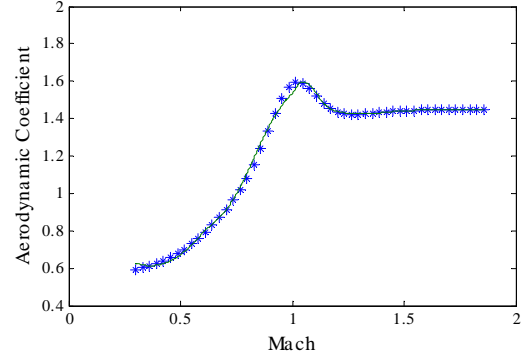
at the second control point. If  $f_1(\bar{q})$  and  $f_2(\bar{q})$  are two local approximations, the weighted global approximation  $F(\bar{q})$  between two control points is given as

$$F(\bar{q}) = W_1(\bar{q})f_1(\bar{q}) + W_2(\bar{q})f_2(\bar{q}).$$

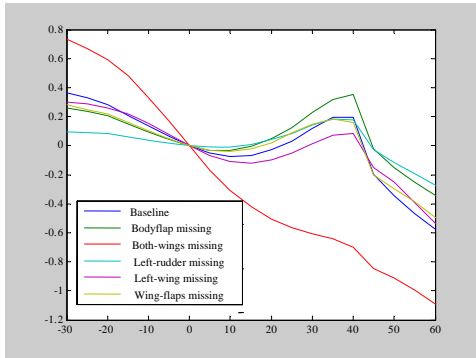
Figure 8 shows the final approximated function that is made of the weighted combinations of the local function approximations.



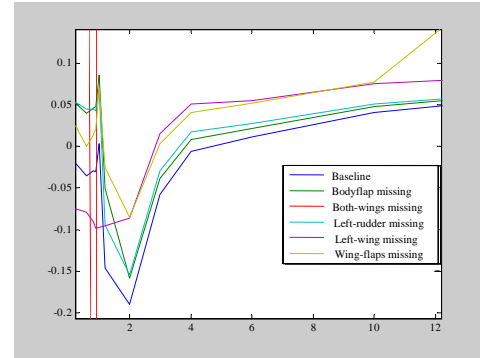
**Figure 7: Local Approximations Centered at Control Points**



**Figure 8: Smooth Functional Approximation using Weighted Local Approximations**



**Figure 9. Moment Coefficient with AOA for Nominal and Various Failure Cases**



**Figure 10. Moment Coefficient with Mach for Nominal and Various Failure Cases**

### 5. Multi-Dimensional Finite Element Approximation

In this section we extend the formulation for FEM modeling for multi dimensional functions. Let  $F$  be given as

$$F = F(q_1, q_2, \dots, q_n), \quad (19)$$

and we have to determine an estimate  $\hat{F}(q_1, q_2, \dots, q_n)$  from a finite element model. As in one-dimensional case, we assume that the domain of  $F$  is covered with finite number of equidistant nodes. Each preliminary local approximation  $F_{i_1, i_2, \dots, i_n}$  is valid in the  $2 \times 2 \times \dots \times 2$  hypercube centered at a node represented by  $n$  indices given as  $(i_1, i_2, \dots, i_n)$ . Each index is numbered in increasing order for the nodes lying in its dimension. Consider a hyper cube formed by  $2^n$  nodes, each node representing a corner of the hyper cube. The final approximation  $F_{i_1, i_2, \dots, i_n}$  is valid in the unit hypercube whose “lower left

corner” is  $(i_1, i_2, \dots, i_n)$ . We shift the origin of the coordinates to the node  $(i_1, i_2, \dots, i_n)$  that has the lowest indices in each dimension. The local coordinates are normalized as

$$\bar{q}_j = \frac{q_j - q_{j,i_j}}{q_{j,i_j+1} - q_{j,i_j}} \quad (20)$$

Let us define a function  $W$  as a function of the normalized coordinate  $\bar{q}_j$  as

$$W(\bar{q}_j) = (1 - \bar{q}_j)^2 (1 + 2\bar{q}_j) \quad (21)$$

Now the weight function for an arbitrary corner node  $(i_1, i_2, \dots, i_k + 1, i_{k+1} + 1, \dots, i_n)$  of the hyper cube can be written as

$$w_{i_1, i_2, \dots, i_k + 1, i_{k+1} + 1, \dots, i_n} = W(\bar{q}_1)W(\bar{q}_2) \cdots W(1 - \bar{q}_k)W(1 - \bar{q}_{k+1}) \cdots W(\bar{q}_n),$$

$$\text{for } 0 \leq \bar{q}_j \leq 1, \quad j = 1 \cdots n \quad (22)$$

The weight function defined by Eq. (22) has the following properties:

- weight is unity at its own node and, it is zero at any other node or edge formed by other nodes.
- summation of all the weights corresponding to the corner nodes of the hyper cube at any point inside the hyper cube is equal to one.

The estimated function  $\hat{F}$  can be defined in terms of normalized coordinates centered at node  $(i_1, i_2, \dots, i_n)$  as

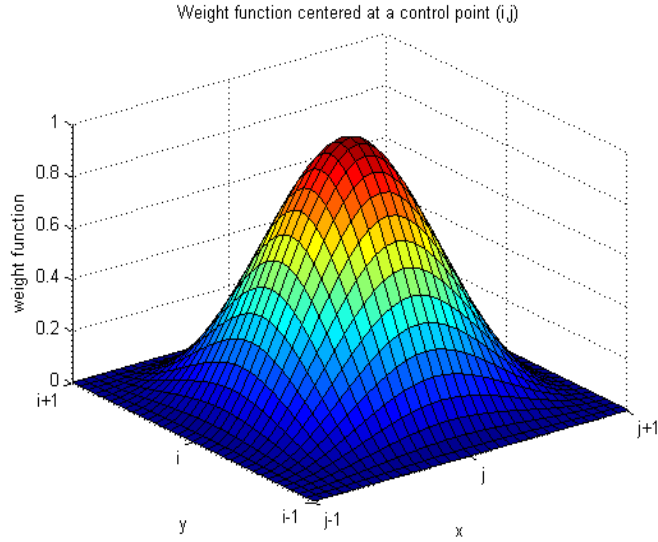
$$\hat{F}(\bar{q}_1, \bar{q}_2, \dots, \bar{q}_n) = \sum_{j=1}^{2^n} w_{i_1, i_2, \dots, i_n, j} F_{i_1, i_2, \dots, i_n, j} \quad (23)$$

#### 6. An Example of Finite Element Function Approximation for Functions of Two Variables.

Let  $Z$  be the matrix of values of a function sampled at the grid points of a 2-d space defined by  $x$  and  $y$ . We are interested in representing the function  $Z$  in closed form, using 2-d polynomial approximation. The closed form representation can be used to evaluate the function  $Z$  at arbitrary  $x$  and  $y$  points. We approximate the function  $Z$  using a number of 2-d polynomial function elements of a smaller area of support. The number of function elements to be used and the degrees of the polynomials in  $x$  and  $y$  are inputs to the algorithm. We specify this by the  $x$  and  $y$  coordinates of the control points as separate vectors. Note that the control points lie exactly on the grid points. At these data points, the approximated value of the function  $Z$  equals the value of the local function centered at that control point. The chosen approximation is locally optimal in the least squared error criteria (i.e., within the area of support of each function element).

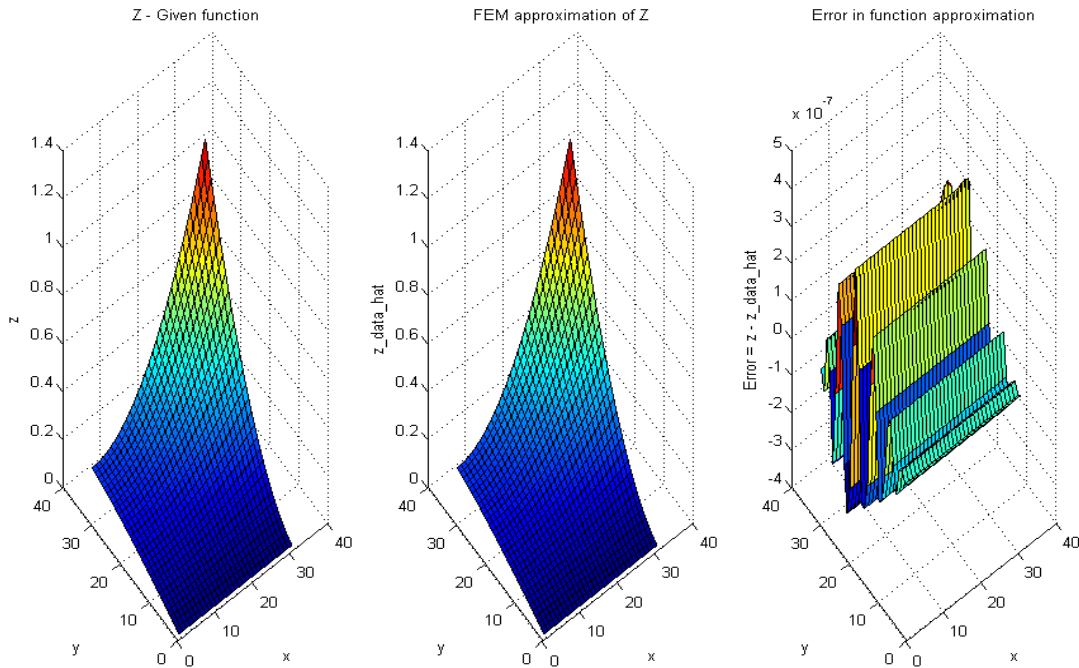
To formulate finite element function approximation satisfying certain constraints, we proceed as follows. The orthonormal basis set of functions are computed that satisfy the constraints and then searched for local functions that are optimal in the vector space spanned by the constrained orthonormal basis functions, based on least square optimality criteria. More details on the finite element function approximation can be found in [7].

Concentrating on the finite element function approximation method for 2-d functions, we initially proceeded by calculating the weights in the  $x$  and  $y$  directions separately. The actual weight function at each point in the grid is the product of the  $x$  and  $y$  weights at that location. The  $x$  and  $y$  weight functions satisfy the criteria that at the current control point location, it has a value of one and gradually decreases to zero as it moves away. The products of the 1-d weight functions also satisfy the same criteria in the 2-d  $x$ - $y$  plane. The intended consequence of this property is that the approximated function value will be equal to the value of the local function at the control point. Figure 11 shows an example of a weight function in the 2-d plane that is the product of the two 1-d weight functions.



**Figure 11. Weight function centered at a control point. The value is one at the location of the control point and gradually decreases and reaches zero at the neighboring control points.**

Figure 12 shows the results of finite element approximation for a surface of  $31 \times 31$  points with  $5 \times 5 = 25$  control points and local polynomials of degree 3 in  $x$  and  $y$ .



**Figure 12. Finite element function approximation. (a) Given function (b) Approximated Function (c) Error between the given and approximated functions. Twenty-five control points were used and each local function is a polynomial of degree 3 in  $x$  and 3 in  $y$ .**

## VII. Conclusion

In this paper we address the function modeling techniques to facilitate an efficient on-line inverse dynamics methodology for trajectory reshaping of the RLVs . On-line trajectory reshaping to determine a feasible reference trajectory is computationally a difficult problem for real time applications.

## Acknowledgments

The research was supported by the AFRL/VACA , WPAFB, Ohio, under contract number FA8650-04-M-3505

## References

- <sup>1</sup> Verma, A., Oppenheimer, M. W. , Doman, D. B., "On-Line Adaptive Estimation and Trajectory Reshaping," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, August 15-18, 2005, San Francisco, CA
- <sup>2</sup> Verma A., Subbarao K. and Junkins, J.L., "A Novel Trajectory Tracking Methodology using Structured Adaptive Model Inversion For UAVs ", 2000 American Control Conference, Chicago, IL, June 28-30, 2000
- <sup>3</sup> Oppenheimer, M.W., Doman, D.B. and Bolander, M.A., "A Method for Estimating Control Failure Effects for Aerodynamic Vehicle Trajectory Retargeting," 2004, AIAA Paper 2004-
- <sup>4</sup> Verma A. and Junkins J.L., "Inverse Dynamics Approach for Real-Time Determination of Feasible Aircraft Reference Trajectories" AIAA Atmospheric Flight Conference, Portland, OR, August 9-11,1999
- <sup>5</sup> Verma A. and Junkins J.L., "Trajectory Generation for Transition from VTOL to Wing-Borne Flight Using Inverse Dynamics" 38th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, January 10-13, 2000
- <sup>6</sup> Bortoff, S.A., "Path Planning for UAVs Using Potential Fields" 2000 American Control Conference, Chicago, IL, June 28-30, 2000
- <sup>7</sup> Verma A., "Determination and Stable Tracking Of Feasible Aircraft Trajectories Using Adaptive Inverse Dynamics", Ph.D. Dissertation, Texas A&M University, May 2000.
- <sup>8</sup> Gao, C., and Hess, R.A., "Inverse Simulation of Large-Amplitude Aircraft Maneuvers," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, 1993, pp. 733-737.
- <sup>9</sup> Junkins, J.L. and Kim, Y., "Stability and Control of Robotic Space Manipulators", *Teleoperation and Robotics in Space*, edited by S. B. Skaar and C. F. Ruoff, Vol.161, Progress in Astronautics and Aeronautics, AIAA, Washington DC, 1995, pp.315-350.
- <sup>10</sup> Fliess, M., Lévine, J., Rouchon, P., "Generalized State Variable Representation for a Simplified Crane Description", *International Journal of Control*, Vol. 58, No.2, 1993, pp. 277-283.
- <sup>11</sup> Nieuwstadt, M.V., Rathinam, M., Murray, R.M., "Differential Flatness and Absolute Equivalence", *Proc. IEEE Control and Decision Conference*, Lake Buena Vista, FL, Dec. 1994, pp.326-333.
- <sup>12</sup> Nieuwstadt, M.V., Murray, R.M., "Real Time Trajectory Generation for Differentially Flat Systems", *International Journal of Robust Nonlinear Control*, Vol.8, Issue 11, September 1998, pp.995-1020.

<sup>13</sup> Schierman, J.D., Hull, J.R., and Ward, D.G., "Adaptive Guidance with Trajectory Reshaping for Reusable Launch Vehicles," Proceedings of the 2002 AIAA Guidance, Navigation, and Control Conference, Monterey, California, August 2002.

<sup>14</sup> Schierman, J.D., Hull, J.R., and D.G. Ward, "On-Line Trajectory Command Reshaping for Reusable Vehicles," Proc. AIAA Guidance, Navigation, and Control Conf., Austin, Texas, Aug. 2003, AIAA Paper No. 2003-5439.

<sup>15</sup> Schierman, J.D., Gandhi, N., Hull, J.R., and D.G. Ward, "Flight Test Results of an Adaptive Guidance System for Reusable Launch Vehicles," Aug. 2004, AIAA Paper No. 2004-4771.

<sup>16</sup> Schierman, J.D., Ward, D.G., Hull, J.R., Gandhi, N., and "Integrated Adaptive Guidance and Control for Re-Entry Vehicles with Flight-Test Results," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, 2004, pp. 975-988.

<sup>17</sup> A A Samarskii, "*Tikhonov's work on computational mathematics*," *Russian Math. Surveys* 22(2) 1967, pp. 149-158

<sup>18</sup> Lancoz, C., "*Linear Differential Operators*," Van Nostrand, London 1961

<sup>19</sup> Tikhonov, A. N., and Arsenin, V. Y., "*Solutions of ill-posed problems*," Winston, Washington DC 1977

<sup>20</sup> Jancaitis, J.R., and Junkins J.L., "Modeling N Dimensional Surfaces Using a Weighting Function Approach," *Journal of Geophysical Research*, Vol. 79, August 1974, pp. 3361.

<sup>21</sup> Junkins J.L., "Investigation of Finite-Element Representations of the Geopotential," *AIAA Journal*, Vol. 14, No. 6, June 1976, pp. 803-808.